


  
 SEARCH

## THE ACM DIGITAL LIBRARY

Feedback

array key Found  
 Terms 11,122  
 used: of  
[array key 253,787](#)

Sort results by relevance

Display results in expanded form

Save Refine these results to a Binder with Advanced Search. Try this search in The ACM Guide.

Open results in a new window

---

Results 1 - 20 of 11,122 Result page: 1 [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

[>>](#)

### [1 Constraint-based array dependence analysis](#)



William Pugh, David Wonnacott

May 1998 ACM Transactions on Programming Languages and Systems (TOPLAS), Volume 20 Issue 3

Publisher: ACM

Full text available: [Pdf \(522.24 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#), [review](#)

**Bibliometrics:** Downloads (6 Weeks): 10, Downloads (12 Months): 67, Citation Count: 13

Traditional array dependence analysis, which detects potential memory aliasing of array references is a key analysis technique for automatic parallelization. Recent studies of benchmark codes indicate that limitations of analysis cause many compilers ...

**Keywords:** Presburger Arithmetic, array dataflow analysis, dependence abstraction, dependence analysis, parallelization, static analysis

## 2 The soft heap: an approximate priority queue with optimal error rate



Bernard Chazelle

November 2000 Journal of the ACM (JACM), Volume 47 Issue 6

**Publisher:** ACM

Full text available: Pdf (112.02 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#), [review](#)

**Bibliometrics:** Downloads (6 Weeks): 13, Downloads (12 Months): 116, Citation Count: 5

A simple variant of a priority queue, called a soft heap, is introduced. The data structure supports the usual operations: insert, delete, meld, and findmin. Its novelty is to beat the logarithmic bound on the complexity of a heap in ...

Keywords: amortization, heap, priority queue, soft heap

## 3 Efficient and precise array access analysis



Yunheung Paek, Jay Hoeflinger, David Padua

January 2002 ACM Transactions on Programming Languages and Systems (TOPLAS), Volume 24 Issue 1

**Publisher:** ACM

Full text available: Pdf (805.10 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 10, Downloads (12 Months): 75, Citation Count: 11

A number of existing compiler techniques hinge on the analysis of array accesses in a program. The most important task in array access analysis is to collect the information about array accesses of interest and summarize it in some standard form. Traditional ...

Keywords: Array access analysis, internal representation, linear memory access descriptors

## 4 Memory safety without garbage collection for embedded applications



Dinakar Dhurjati, Sumant Kowshik, Vikram Adve, Chris Lattner

February 2005 ACM Transactions on Embedded Computing Systems (TECS), Volume 4 Issue 1

**Publisher:** ACM

Full text available: Pdf (511.25 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 4, Downloads (12 Months): 99, Citation Count: 8

Traditional approaches to enforcing memory safety of programs rely heavily on run-time checks of memory accesses and on garbage collection, both of which are unattractive for embedded applications. The goal of our work is to develop advanced compiler ...

Keywords: Embedded systems, automatic pool allocation, compilers, programming languages, region management, security, static analysis

## 5 A new group rekeying scheme based on t-packing designs for ad hoc networks

Jianwei Chen, Li Xu, Yi Mu

June InfoScale '07: Proceedings of the 2nd international conference on Scalable  
2007 information systems

**Publisher:** ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications  
Engineering)

Full text available:  Pdf (136.35 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 11, Downloads (12 Months): 20, Citation Count: 0

Secure group key distribution and efficient rekeying is one of the most challenging security issues in ad hoc networks at present. In this paper, Latin squares are used to construct orthogonal arrays in order to quickly obtain t-packing designs. Based ...

Keywords: ad hoc network, cover-free family, group rekeying, packing design

## 6 A study of Erlang ETS table implementations and performance

Scott Lystig Fritchie

August ERLANG '03: Proceedings of the 2003 ACM SIGPLAN workshop on Erlang  
2003

**Publisher:** ACM

Full text available:  Pdf (232.17 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 6, Downloads (12 Months): 73, Citation Count: 1

The viability of implementing an in-memory database, Erlang ETS, using a relatively-new data structure, called a Judy array, was studied by comparing the performance of ETS tables based on four data structures: AVL balanced binary trees, B-trees, resizable ...

Keywords: AVL tree, B-tree, Erlang, Judy array, hash table, in-memory database

## 7 Security on FPGAs: State-of-the-art implementations and attacks

Thomas Wollinger, Jorge Guajardo, Christof Paar

August ACM Transactions on Embedded Computing Systems (TECS), Volume 3 Issue  
2004 3

**Publisher:** ACM

Full text available:  Pdf (296.79 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 41, Downloads (12 Months): 427, Citation Count: 3

In the last decade, it has become apparent that embedded systems are integral parts of our every day lives. The wireless nature of many embedded applications as well as their omnipresence has made the need for security and privacy preserving mechanisms ...

Keywords: Cryptography, FPGA, attacks, cryptographic applications, reconfigurable hardware, reverse engineering, security

## 8 Tiling optimizations for 3D scientific computations

Gabriel Rivera, Chau-Wen Tseng

November 2000 Supercomputing '00: Proceedings of the 2000 ACM/IEEE conference on Supercomputing (CDROM)

**Publisher:** IEEE Computer Society

Full text available:  Publisher Site ,  Pdf (318.48 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 2, Downloads (12 Months): 33, Citation Count: 12

Compiler transformations can significantly improve data locality for many scientific programs. In this paper, we show that iterative solvers for partial differential equations (PDEs) in three dimensions require new compiler optimizations not needed for ...

## 9 A Multi-Platform Co-Array Fortran Compiler

Yuri Dotsenko, Cristian Coarfa, John Mellor-Crummey

September 2004 PACT '04: Proceedings of the 13th International Conference on Parallel Architectures and Compilation Techniques

**Publisher:** IEEE Computer Society

Full text available:  Pdf (269.77 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#)

**Bibliometrics:** Downloads (6 Weeks): 2, Downloads (12 Months): 19, Citation Count: 9

Co-array Fortran (CAF)-a small set of extensions to Fortran 90-is an emerging model for scalable, global address space parallel programming.CAF's global address space programming model simplifies the development of single-program-multiple-data parallel ...

## 10 Scratchpad allocation for data aggregates in superperfect graphs

Lian Li, Quan Hoang Nguyen, Jingling Xue

June 2007 LCTES '07: Proceedings of the 2007 ACM SIGPLAN/SIGBED conference on Languages, compilers, and tools for embedded systems

**Publisher:** ACM

Full text available:  Pdf (313.21 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 6, Downloads (12 Months): 79, Citation Count: 0

Existing methods place data or code in scratchpad memory, i.e., SPM by either relying on heuristics or resorting to integer programming or mapping it to a graph coloring problem. In this work, the SPM allocation problem is formulated as an interval coloring ...

**Keywords:** SPM, SPM allocation, interval coloring, scratchpad memory, superperfect graph

## 11 Scratchpad allocation for data aggregates in superperfect graphs

Lian Li, Quan Hoang Nguyen, Jingling Xue  
July 2007 ACM SIGPLAN Notices, Volume 42 Issue 7

**Publisher:** ACM

Full text available:  Pdf (313.21 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 6, Downloads (12 Months): 79, Citation Count: 0

Existing methods place data or code in scratchpad memory, i.e., SPM by either relying on heuristics or resorting to integer programming or mapping it to a graph coloring problem. In this work, the SPM allocation problem is formulated as an interval coloring ...

**Keywords:** SPM, SPM allocation, interval coloring, scratchpad memory, superperfect graph

## 12 A checkpointing strategy for scalable recovery on distributed parallel systems

Vijay K. Naik, Samuel P. Midkiff, Jose E. Moreira  
November 1997 Supercomputing '97: Proceedings of the 1997 ACM/IEEE conference on Supercomputing (CDROM)

**Publisher:** ACM

Full text available:  Pdf (144.90 KB)

Additional Information: [full citation](#), [abstract](#), [references](#)

**Bibliometrics:** Downloads (6 Weeks): 4, Downloads (12 Months): 44, Citation Count: 0

In this paper, we describe a new scheme for checkpointing parallel applications on message-passing scalable distributed memory systems. The novelty of our scheme is that a checkpointed application can be restored, from its checkpointed state, in a reconfigured ...

**Keywords:** DRMS, IBM RS/6000 SP, checkpointing and restart, parallel checkpointing, reconfigurable checkpointing, scalable recovery

## 13 Ensuring code safety without runtime checks for real-time control systems

Sumant Kowshik, Dinakar Dhurjati, Vikram Adve  
October 2002 CASES '02: Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems

**Publisher:** ACM

Full text available:  Pdf (127.10 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 5, Downloads (12 Months): 42, Citation Count: 8

This paper considers the problem of providing safe programming support and enabling secure online software upgrades for control software in real-time control systems. In such systems, offline techniques for ensuring code safety are greatly preferable ...

**Keywords:** compiler, control, programming language, real-time, security, static analysis

## 14 Memory safety without runtime checks or garbage collection

 Dinakar Dhurjati, Sumant Kowshik, Vikram Adve, Chris Lattner

July 2003 ACM SIGPLAN Notices, Volume 38 Issue 7

**Publisher:** ACM

Full text available:  Pdf (245.47 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 3, Downloads (12 Months): 63, Citation Count: 10

Traditional approaches to enforcing memory safety of programs rely heavily on runtime checks of memory accesses and on garbage collection, both of which are unattractive for embedded applications. The long-term goal of our work is to enable 100% static ...

**Keywords:** automatic pool allocation, compilers, embedded systems, programming languages, region management, security, static analysis

## 15 Memory safety without runtime checks or garbage collection

 Dinakar Dhurjati, Sumant Kowshik, Vikram Adve, Chris Lattner

June 2003 LCTES '03: Proceedings of the 2003 ACM SIGPLAN conference on Language, compiler, and tool for embedded systems

**Publisher:** ACM

Full text available:  Pdf (245.47 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 3, Downloads (12 Months): 63, Citation Count: 10

Traditional approaches to enforcing memory safety of programs rely heavily on runtime checks of memory accesses and on garbage collection, both of which are unattractive for embedded applications. The long-term goal of our work is to enable 100% static ...

**Keywords:** automatic pool allocation, compilers, embedded systems, programming languages, region management, security, static analysis

## 16 Arrays of objects

 Morten J. Kromberg

October 2007 DLS '07: Proceedings of the 2007 symposium on Dynamic languages

**Publisher:** ACM

Full text available:  Pdf (246.71 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 0, Downloads (12 Months): 55, Citation Count: 0

This paper discusses key design decisions faced by a language design team while adding Object Oriented language features to Dyalog, a modern dialect of APL. Although classes and interfaces are first-class language elements in the new language, and arrays ...

**Keywords:** arrays, functional programming, language design, multi-paradigm languages, object orientation

**17 Cache-efficient string sorting using copying**

Ranjan Sinha, Justin Zobel, David Ring

February 2007 Journal of Experimental Algorithmics (JEA), Volume 11

**Publisher:** ACM

Full text available:  Pdf (1.92 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 21, Downloads (12 Months): 240, Citation Count: 1

Burstsort is a cache-oriented sorting technique that uses a dynamic trie to efficiently divide large sets of string keys into related subsets small enough to sort in cache. In our original burstsort, string keys sharing a common prefix were managed via ...

**Keywords:** Sorting, algorithms, cache, experimental algorithms, string management, tries

**18 Compact dictionaries for variable-length keys and data with applications**

Daniel K. Blandford, Guy E. Blelloch

May 2008 ACM Transactions on Algorithms (TALG), Volume 4 Issue 2

**Publisher:** ACM

Full text available:  Pdf (256.44 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 16, Downloads (12 Months): 173, Citation Count: 0

We consider the problem of maintaining a dynamic dictionary  $T$  of keys and associated data for which both the keys and data are bit strings that can vary in length from zero up to the length  $w$  of a machine word. We present a data structure ...

**Keywords:** Compression

**19 Automatic data layout for distributed-memory machines**

Ken Kennedy, Ulrich Kremer

July ACM Transactions on Programming Languages and Systems (TOPLAS), Volume 1998 20 Issue 4

**Publisher:** ACM

Full text available:  Pdf (633.20 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [cited by](#), [index terms](#), [review](#)

**Bibliometrics:** Downloads (6 Weeks): 7, Downloads (12 Months): 64, Citation Count: 23

The goal of languages like Fortran D or High Performance Fortran (HPF) is to provide a simple yet efficient machine-independent parallel programming model. After the algorithm selection, the data layout choice is the key intellectual challenge in writing ...

**Keywords:** high performance Fortran

## 20 No sorting? better searching!



Gianni Franceschini, Roberto Grossi

March 2008 ACM Transactions on Algorithms (TALG), Volume 4 Issue 1

Publisher: ACM

Full text available: Pdf (145.86 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

**Bibliometrics:** Downloads (6 Weeks): 23, Downloads (12 Months): 222, Citation Count: 0

Questions about order versus disorder in systems and models have been fascinating scientists over the years. In computer science, order is intimately related to sorting, commonly meant as the task of arranging keys in increasing or decreasing order with ...

**Keywords:** Implicit data structures, in-place algorithms, searching, sorting

---

Results 1 - 20 of 11,122 Result page: 1 2 3 4 5 6 7 8 9 10 next

[>>](#)

The ACM Portal is

published by the Association for Computing Machinery. Copyright © 2008 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads: [Adobe Acrobat](#) [QuickTime](#) [Windows Media Player](#) [Real Player](#)



## Advanced Search

## THE ACM DIGITAL LIBRARY

No results were found.

Please revise or [start a new search instead](#)

**Enter words, phrases or names below to revise your search.**

Surround phrases or full names with double quotation marks.

Edit the query directly, or use the form below

ephemeral, key, symmetric, sub-array, array

## Words or Phrases

Find  [any field] with

**all** of this text (and)

**any** of this text (or)

**none** of this text (not)

## Names

Find  [any field] with  
names

using  all  any  none of the names

## Keywords

Find author's  
keywords

using  all  any  none of the  
keywords

## Affiliations

Find company or  
school

using  all  any  
 none of the affiliations

## Publication

Find  
publication

using  all  any  none of the  
names

Find  
publisher

using  any  none of the names

## Published since

[year]

Published before  [year]

In publication  
types

Journal  Proceeding  Transaction  Magazine  
 Newsletter

## Conference

Find sponsor  
names

using  all  any  none of the  
names

Find  
location

using  any  none of the locations

Find year (yyyy)

using  any  none of the years

## Identification codes

Find ISBN/  
ISSN

Find  
DOI

## Computing Classification System (CCS)

Find node

Find subject/  
noun

Look at primary category only

## Required components

Results must have  Full Text  Abstract  
 Review



The ACM Portal is published by the Association for Computing Machinery. Copyright © 2008 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)



## Advanced Search

## THE ACM DIGITAL LIBRARY

No results were found.

Please revise or [start a new search instead](#)

**Enter words, phrases or names below to revise your search.**

Surround phrases or full names with double quotation marks.

Edit the query directly, or use the form below

ephemeral, key, symmetric, sub-array, array

## Words or Phrases

Find  [any field] with

**all** of this text (and)

**any** of this text (or)

**none** of this text (not)

## Names

Find  [any field] with  
names

using  all  any  none of the names

## Keywords

Find author's  
keywords

using  all  any  none of the  
keywords

## Affiliations

Find company or  
school

using  all  any  
 none of the affiliations

## Publication

Find  
publication

using  all  any  none of the  
names

Find  
publisher

using  any  none of the names

## Published since

[year]

Published before  [year]

In publication  
types

- Journal
- Proceeding
- Transaction
- Magazine
- Newsletter

## Conference

Find sponsor  
names

using  all  any  none of the  
names

Find  
location

using  any  none of the locations

Find year (yyyy)

using  any  none of the years

## Identification codes

Find ISBN/  
ISSN

Find  
DOI

## Computing Classification System (CCS)

Find node

Find subject/  
noun

Look at primary category only

## Required components

Results must have  Full Text  Abstract  
 Review



The ACM Portal is published by the Association for Computing Machinery. Copyright © 2008 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)